UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/735,026	12/12/2003	James Blair Chapman	11235	2461

7590 05/31/2006

John D. Cowart
Teradata Law IP, WHQ-4W
NCR Corporation
1700 S. Patterson Blvd.
Dayton, OH 45479-0001

EXAMINER	
SOMMERFELD, PAUL J	

ART UNIT	PAPER NUMBER
2168	

DATE MAILED: 05/31/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	10/735,026	CHAPMAN, JAMES BLAIR	
	Examiner	Art Unit	
	Paul J. Sommerfeld	2168	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *12 December 2003*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-26* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-26* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *12 December 2003* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

Claim Objections

1. Claim 26 is objected to because of the following informalities:

On line 4 of claim 26, "providing access to on or more" should be replaced by --providing access to one or more--. This appears to be a typographical error.

Appropriate correction is required.

Claim Rejections - 35 USC § 112

1. The following is a quotation of the first paragraph of 35 U.S.C. 112:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

1. Claim 26 is rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

Claim 26 recites "a set of one or more database tables residing on the one or more nodes, the one or more database tables containing information organized by geographic location". Although paragraph [0016] of the specification of the instant invention describes tables stored across multiple data-storage facilities, it does not

describe that the tables contain information organized by geographic location. Therefore, claim 26 contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

2. The following is a quotation of the second paragraph of 35 U.S.C. 112:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

3. Claims 18, 25, and 26 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

Claim 18 recites a "computer executable code for instructing a computer to" perform steps of the method of the instant invention. The recited language "code for" makes it unclear as to whether the steps that the claimed computer executable code is for are limitations in claim 26, or are simply optional. See MPEP § 2111.04.

For purposes of examination, the recited "computer executable code" is interpreted as any code, instructions, program, or software executable by a computer.

Claim 25 recites "a memory operable to store a program". The recited language "operable to" makes it unclear as to whether the steps the memory is operable to perform are limitations in the claim, or are simply optional. See MPEP § 2111.04.

For purposes of examination, "a memory operable to store a program" is interpreted as any type of memory which is capable of having stored therein code, instructions, a program, or software.

Claim 25 recites "at least one processor operable to determine". The recited language "operable to" makes it unclear as to whether the steps the processor is operable to perform are limitations in the claim, or are simply optional. See MPEP § 2111.04.

For purposes of examination, "at least one processor operable to determine" is interpreted as any processor capable of executing instructions.

Claim 25 recites "the at least one processor capable of". The recited language "capable of" makes it unclear as to whether the steps the processor is capable of performing are limitations in the claim, or are simply optional. See MPEP § 2111.04.

For purposes of examination, "the at least one processor capable of" is interpreted as any processor capable of executing instructions.

Claim 26 recites a "plurality of virtual processes operable to" perform steps of the method of the present invention. The recited language "operable to" makes it unclear as to whether the steps that the claimed virtual processes are operable to perform are limitations in claim 26, or are simply optional. See MPEP § 2111.04.

For purposes of examination, "plurality of virtual processes operable to" is interpreted as any code, instructions, program, or software executing on a computer.

Claim Rejections - 35 USC § 102

4. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

5. Claim 25 is rejected under 35 U.S.C. 102(b) as being anticipated by Chen et al (U.S. Patent Number 5,819,083).

As to claim 25, Chen et al teaches a relational database management system (see Abstract), comprising:

a memory operable to store a program accessible to one or more of a plurality of transmitting modules, the program capable of managing a redistribution of one or more rows associated with one or more database tables (item 25 in Figure 2, showing the memory of the computer system in Figure 2); and

at least one processor operable to determine the number of transmitting modules on which the program was invoked, the at least one processor capable of executing a few-rows row redistribution method to redistribute the one or more rows if the program was invoked on a singe transmitting module, the at least one processor also capable of

executing a many-rows row redistribution method to redistribute the one or more rows (item 22 in Figure 2, showing the processor of the computer system in Figure 2).

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

7. Claims 1-24 and 26 are rejected under 35 U.S.C. 103(a) as being unpatentable over Chen et al (U.S. Patent Number 5,819,083) and further in view of Shatdal et al (Shatdal, A., Naughton, J. F., "Adaptive Parallel Aggregation Algorithms", 1995, SIGMOD '95, pp. 104-114).

As to claim 1, Chen et al teaches a method for redistributing data in a relational data base management system (see Abstract), comprising:

allocating a buffer associated with a transmitting processing module, the transmitting processing module having access to a program, the program capable of managing a redistribution of one or more rows associated with one or more database tables (col. 2 lines 47-50, allocating a communication buffer for transferring rows between nodes, enabling redistribution of rows);

if the allocated buffer is larger than the portion of the buffer to be occupied by the one or more rows:

storing one or more rows of a database table in the allocated buffer (col. 5 lines 34-37, indicating that portions of database tables are stored in the buffer);

communicating a message to one or more destination processing modules, the message comprising at least some of the one or more rows stored in the allocated buffer (col. 6 lines 48-51, sending row data contained in a buffer to a node);

Chen et al does not explicitly teach comparing the allocated buffer to a portion of the buffer to be occupied by the one or more rows;

executing a many-rows method to redistribute the one or more rows.

Shatdal et al teaches comparing the allocated buffer to a portion of the buffer to be occupied by the one or more rows (p. 106 col. 1 lines 12-16, determining if a table will fit in allocated memory);

executing a many-rows method to redistribute the one or more rows (p. 106 col. 1 lines 12-16, the allocated memory is not large enough to fit the table, tuples (i.e. rows) are partitioned into multiple buckets. This is a many-rows method of redistributing, since it is executed when there the number of rows is too large to fit into allocated memory.).

Therefore, it would have been obvious to one skilled in the art at the time the invention was made to have modified the method of redistributing rows in a database taught by Chen et al by the step of comparing the size of a buffer to the size of a table

of rows to be stored in the buffer, because comparing the size of a buffer to the size of a table of rows to be stored in the buffer while redistributing rows in a database enables dynamic adaptation to grouping selectivities at query evaluation time (Shatdal et al p. 104 col. 1 lines 7-13).

As to claims 2 and 12, Chen et al, as modified by Shatdal et al, teaches the message comprises all of the one or more rows stored in the allocated buffer (Chen et al col. 6 lines 48-49).

As to claims 3, 13, and 19, Chen et al, as modified by Shatdal et al, teaches the transmitting processing module comprises one of a plurality of processing modules associated with a relational database system (Chen et al col. 5 lines 34-37, indicating each node has a transportation layer containing a communication buffer).

As to claims 4, 14, and 20, Chen et al, as modified by Shatdal et al, teaches the message is communicated to each of a plurality of destination processing elements (Chen et al col. 6 lines 59-60).

As to claims 5 and 21, Chen et al, as modified by Shatdal et al, teaches the allocated buffer is capable of storing no more than ten (10) rows (Chen et al col. 8 lines 50-54, since the parameters of the expression can be varied to produce an arbitrary

value, and since the expression determines the size of the buffer, the invention is capable of allocating a buffer capable of storing no more than ten (10) rows.).

As to claims 6, 16, and 22, Chen et al, as modified by Shatdal et al, teaches the many-rows row redistribution method comprises:

communicating from one or more transmitting modules a first signal to a plurality of processing modules within a relational database system, the first signal operable to initiate a row receiver task on each of the processing modules (Chen et al col. 5 lines 34-37);

communicating from one or more of the processing modules a ready-to-receive signal to the one or more transmitting modules (Chen et al item 152 in Figure 6, col. 6 lines 10-13);

communicating from the one or more transmitting modules a second signal comprising the one or more rows associated with the database table (Chen et al col. 6 lines 48-51);

after communication of the last row associated with the database table, communicating from the one or more transmitting modules an end-of-data signal to each of the plurality of processing modules (Chen et al col. 6 lines 53-55 and 59-60).

As to claims 7 and 23, Chen et al, as modified by Shatdal et al, teaches invoking the program on a single transmitting processing module (Chen et al col. 6 lines 46-47, initiating the program).

As to claim 8, Chen et al, as modified by Shatdal et al, teaches receiving at each of a plurality of destination processing elements a substantially similar set of the one or more rows stored in the allocated buffer (col. 6 lines 48-51, where the rows are substantially similar in that they are contained in the same buffer).

As to claims 9, 15, and 24, Chen et al, as modified by Shatdal et al, teaches determining a number of rows to store in the allocated buffer (Chen et al col. 8 lines 51-54, determining the size of the buffer inherently determines the number of rows that can be stored in the buffer).

As to claim 10, Chen et al teaches a method for redistributing data in a relational data base management system (see Abstract), comprising:

invoking a program on one or more of a plurality of transmitting modules, the program capable of managing a redistribution of one or more rows associated with one or more database tables (col. 2 lines 47-50, redistributing rows on a plurality of nodes);

Chen et al does not explicitly teach if the program was invoked on a single transmitting module:

executing a few-rows redistribution method to redistribute the one or more rows;

otherwise:

executing a many-rows redistribution method to redistribute the one or more rows.

Shatdal et al teaches if the program was invoked on a single transmitting module:

executing a few-rows redistribution method to redistribute the one or more rows (p. 106 col. 1 lines 12-16, if it was not determined that the table is too large to fit in allocated memory, no additional partitioning is done. This is a few-rows method, since it is executed when the number of rows is small enough to fit into allocated memory.);

otherwise:

executing a many-rows redistribution method to redistribute the one or more rows (p. 106 col. 1 lines 12-16, the allocated memory is not large enough to fit the table, tuples (i.e. rows) are partitioned into multiple buckets. This is a many-rows method of redistributing, since it is executed when there the number of rows is too large to fit into allocated memory.).

Therefore, it would have been obvious to one skilled in the art at the time the invention was made to have modified the method of redistributing rows in a database taught by Chen et al by the step of comparing the size of a buffer to the size of a table of rows to be stored in the buffer, because comparing the size of a buffer to the size of a table of rows to be stored in the buffer while redistributing rows in a database enables dynamic adaptation to grouping selectivities at query evaluation time (Shatdal et al p. 104 col. 1 lines 7-13).

As to claim 11, Chen et al, as modified by Shatdal et al, teaches the few-rows row redistribution method comprises:

allocating a buffer associated with a transmitting processing module, the transmitting processing module having access to the program, the program associated with a single transmitting module (Chen et al col. 2 lines 47-50, allocating a communication buffer for transferring rows between nodes, enabling redistribution of rows);

comparing the allocated buffer to a portion of the buffer to be occupied by the one or more rows (Shatdal et al p. 106 col. 1 lines 12-16, determining if a table will fit in allocated memory);

if the allocated buffer is larger than the portion of the buffer to be occupied by the one or more rows:

storing one or more rows of a database table in the allocated buffer (Chen et al col. 5 lines 34-37, indicating that portions of database tables are stored in the buffer);

communicating a message to one or more destination modules, the message comprising at least some of the one or more rows stored in the allocated buffer (Chen et al col. 6 lines 48-51, sending row data contained in a buffer to a node);

otherwise:

executing a many-rows method to redistribute the one or more rows (Shatdal et al p. 106 col. 1 lines 12-16, the allocated memory is not large enough

to fit the table, tuples (i.e. rows) are partitioned into multiple buckets. This is a many-rows method of redistributing, since it is executed when there the number of rows is too large to fit into allocated memory.).

As to claim 17, Chen et al, as modified by Shatdal et al, teaches determining the number of transmitting modules on which the program was invoked (Chen et al col. 6 lines 59-60, determining the number of transmitting modules is inherent, since each module is notified that reception of the information is complete.).

As to claim 18, Chen et al teaches a computer-readable medium containing computer-executable code (col. 4 lines 46-49) for instructing a computer to:

allocate a buffer associated with a transmitting processing module, the transmitting processing module having access to a program, the program capable of managing a redistribution of one or more rows associated with one or more database tables (col. 2 lines 47-50, allocating a communication buffer for transferring rows between nodes, enabling redistribution of rows);

if the allocated buffer is larger than the portion of the buffer to be occupied by the one or more rows:

store one or more rows associated with a database table in the allocated buffer (col. 5 lines 34-37, indicating that portions of database tables are stored in the buffer);

communicate a message to one or more destination processing modules, the message comprising at least some of the one or more rows stored in the allocated buffer (col. 6 lines 48-51, sending row data contained in a buffer to a node);

Chen et al does not explicitly teach compare the allocated buffer to a portion of the buffer to be occupied by the one or more rows;

execute a many-rows method to redistribute the one or more row.

Shatdal et al teaches compare the allocated buffer to a portion of the buffer to be occupied by the one or more rows (p. 106 col. 1 lines 12-16, determining if a table will fit in allocated memory);

execute a many-rows method to redistribute the one or more rows (p. 106 col. 1 lines 12-16, the allocated memory is not large enough to fit the table, tuples (i.e. rows) are partitioned into multiple buckets. This is a many-rows method of redistributing, since it is executed when there the number of rows is too large to fit into allocated memory.).

Therefore, it would have been obvious to one skilled in the art at the time the invention was made to have modified the method of redistributing rows in a database taught by Chen et al by the step of comparing the size of a buffer to the size of a table of rows to be stored in the buffer, because comparing the size of a buffer to the size of a table of rows to be stored in the buffer while redistributing rows in a database enables dynamic adaptation to grouping selectivities at query evaluation time (Shatdal et al p. 104 col. 1 lines 7-13).

As to claim 26, Chen et al teaches a database management system (Chen et al Figure 2, col. 3, line 10), comprising:

a massively parallel processing system (Figure 2, col. 3, line 10) comprising:

one or more nodes (items 61, 63, 65, and 67, Figure 2);

a plurality of processors, each of the one or more nodes providing access to on or more processors (item 22 in Figure 2, showing a processor in a node); and

a plurality of virtual processes, each of the one or more processors providing access to one or more virtual processes (col. 4 lines 42-45);

a set of one or more database tables containing information organized by geographic location (items 62, 66, and 64 in Figure 3, col. 5 lines 5-6); and

one or more of the plurality of virtual processes operable to:

allocate a buffer associated with a transmitting processing module, the transmitting processing module having access to a program, the program capable of managing a redistribution of one or more rows associated with one or more database tables (col. 2 lines 47-50, allocating a communication buffer for transferring rows between nodes, enabling redistribution of rows);

if the allocated buffer is larger than the portion of the buffer to be occupied by the one or more rows:

store one or more rows associated with a database table in the allocated buffer (col. 5 lines 34-37, indicating that portions of database tables are stored in the buffer);

communicate a message to one or more destination processing modules, the message comprising at least some of the one or more rows stored in the allocated buffer (col. 6 lines 48-51, sending row data contained in a buffer to a node);

Chen et al does not explicitly teach compare the allocated buffer to a portion of the buffer to be occupied by the one or more rows compare the allocated buffer to a portion of the buffer to be occupied by the one or more rows;

execute a many-rows method to redistribute the one or more rows.

Shatdal et al teaches compare the allocated buffer to a portion of the buffer to be occupied by the one or more rows compare the allocated buffer to a portion of the buffer to be occupied by the one or more rows (p. 106 col. 1 lines 12-16, determining if a table will fit in allocated memory);

execute a many-rows method to redistribute the one or more rows (p. 106 col. 1 lines 12-16, the allocated memory is not large enough to fit the table, tuples (i.e. rows) are partitioned into multiple buckets. This is a many-rows method of redistributing, since it is executed when there the number of rows is too large to fit into allocated memory.).

Therefore, it would have been obvious to one skilled in the art at the time the invention was made to have modified the method of redistributing rows in a database

taught by Chen et al by the step of comparing the size of a buffer to the size of a table of rows to be stored in the buffer, because comparing the size of a buffer to the size of a table of rows to be stored in the buffer while redistributing rows in a database enables dynamic adaptation to grouping selectivities at query evaluation time (Shatdal et al p. 104 col. 1 lines 7-13).

Conclusion

8. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure.

- U.S. Patent Number 6,496,823 B2, issued to Blank et al, for teaching a method of apportioning work to processors in a multi-processor system.
- U.S. Publication 2001/0047360 A1, issued to Huras et al, for teaching a method of reorganizing a database.
- Luo, G., Ellman, C. J., Haas, P. J., Naughton, J. F., "A Scalable Ripple Join Algorithm", ACM SIGMOD 2002, pp. 252-262, for teaching a method of redistributing rows in a parallel database.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Paul J. Sommerfeld whose telephone number is 571 272-6545. The examiner can normally be reached on M-F 7:45 am - 4:15pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tim T. Vo can be reached on 571 272-3642. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

TIM VO
PRIMARY EXAMINER